



186.172 Algorithmen und Datenstrukturen 1 VL 4.0

2. Übungstest SS 2008

6. Juni 2008

Machen Sie die folgenden Angaben bitte in deutlicher Blockschrift:

Nachname: Vorname:

Matrikelnummer: Studienkennzahl:

Anzahl abgegebener Zusatzblätter:

Legen Sie bitte Ihren Studentenausweis vor sich auf das Pult.

Sie können die Lösungen entweder direkt auf die Angabeblätter oder auf Zusatzblätter schreiben, die Sie auf Wunsch von der Aufsicht erhalten. Es ist nicht zulässig, eventuell mitgebrachtes eigenes Papier zu verwenden.

Die Verwendung von Taschenrechnern, Mobiltelefonen, Skripten, Büchern, Mitschriften, Ausarbeitungen oder vergleichbaren Hilfsmitteln ist unzulässig.

Die Arbeitszeit beträgt 55 Minuten.

	A1:	A2:	A3:	Summe:
Erreichbare Punkte:	16	16	18	50
Erreichte Punkte:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Viel Erfolg!

Aufgabe 1.A: Hashverfahren

(16 Punkte)

a) (7 Punkte)

Gegeben ist eine Hashtabelle mit Tabellengröße $m = 13$ und der Hashfunktion $h'(k) = k \bmod 13$. Zur Kollisionsbehandlung wird quadratisches Sondieren mit den Konstanten $c_1 = 3$ und $c_2 = 2$ verwendet.

- Fügen Sie den Wert 18 in die folgende Tabelle ein:

0	1	2	3	4	5	6	7	8	9	10	11	12
				30		32	20					25

- Fügen Sie den Wert 17 in die folgende Tabelle ein:

0	1	2	3	4	5	6	7	8	9	10	11	12
				30		32	20					25

- Fügen Sie den Wert 7 in die folgende Tabelle ein:

0	1	2	3	4	5	6	7	8	9	10	11	12
				30		32	20					25

b) (9 Punkte)

Gegeben ist eine Hashtabelle mit Tabellengröße $m = 7$, die Double Hashing mit der Verbesserung nach Brent benutzt mit

$$h_1(k) = k \bmod 7$$

$$h_2(k) = (2k + 1) \bmod 5$$

- Fügen Sie den Wert 21 in die folgende Tabelle ein:

0	1	2	3	4	5	6
14			24		19	

- Fügen Sie den Wert 5 in die folgende Tabelle ein:

0	1	2	3	4	5	6
14			24		19	

- Welches Problem kann beim Einfügen in die Hashtabelle bei der Verwendung der oben angegebenen Hashfunktionen auftreten? Wie müssen die Funktionen $h_1(k)$ und/oder $h_2(k)$ abgeändert werden, um dieses Problem zu beheben?

Aufgabe 2.A: Suchverfahren

(16 Punkte)

a) (12 Punkte)

- Schreiben Sie detaillierten Pseudocode für eine effiziente rekursive Prozedur, die die Höhe $x.height$ und die Balancierung $x.bal$ jedes Knotens x eines natürlichen binären Suchbaumes T berechnet. Geben Sie unter Umständen notwendige Initialisierungen an sowie den ersten Aufruf mit allen Parametern, um die Rekursion zu starten. Auf die Wurzel des Baumes kann mithilfe von T zugegriffen werden. Zu jedem Knoten x werden folgende Daten gespeichert:

$x.key$: Schlüssel von x
 $x.left$: Verweis auf linkes Kind von x
 $x.right$: Verweis auf rechtes Kind von x
 $x.height$: Höhe des Teilbaumes mit Wurzel x
 $x.bal$: Balancierung des Teilbaumes mit Wurzel x

Hinweis: Die Werte $x.height$ und $x.bal$ sind zu Beginn nicht initialisiert, sondern sollen von Ihrer Prozedur berechnet werden!

- Geben Sie die Laufzeit ihres Algorithmus in Θ -Notation in Abhängigkeit der Anzahl der Knoten n an.
- Auf welchem aus der Vorlesung bekannten allgemeinen Verfahren basiert ihre Implementierung?

b) (4 Punkte)

- Fügen Sie folgende Werte in genau dieser Reihenfolge in einen Anfangs leeren natürlichen binären Suchbaum ein. Zeichnen Sie lediglich den resultierenden Baum, Zwischenschritte müssen nicht angegeben werden.

$\langle 34, 15, 50, 7, 30, 40, 80, 3, 10, 20, 33, 38, 99 \rangle$

- Handelt es sich bei dem resultierenden Baum um einen gültigen AVL-Baum? Begründen Sie Ihre Antwort.

Aufgabe 3.A: Graphen

(18 Punkte)

Gegeben ist die folgende Adjazenzmatrix eines ungerichteten Graphen G :

	A	B	C	D	E	F	G	H	I	J				
A	0	0	0	0	0	0	1	0	0	1	(A)	(J)	(E)	(D)
B	0	0	0	1	0	0	0	0	0	0				
C	0	0	0	0	0	1	0	0	1	0				
D	0	1	0	0	0	0	0	0	0	0	(G)	(H)	(B)	(C)
E	0	0	0	0	0	0	0	1	0	1				
F	0	0	1	0	0	0	0	0	1	0				
G	1	0	0	0	0	0	0	0	0	1			(F)	(I)
H	0	0	0	0	1	0	0	0	0	1				
I	0	0	1	0	0	1	0	0	0	0				
J	1	0	0	0	1	0	1	1	0	0				

a) (4 Punkte)

Zeichnen Sie den Graphen G in die gegebene Vorlage ein und geben Sie G in Adjazenzlistendarstellung an. Kreuzen Sie außerdem die zutreffenden Eigenschaften des Graphen G an:

ist Baum ist Wald ist zusammenhängend enthält Kreis

b) (4 Punkte)

Ein ungerichteter Graph heißt *bipartit*, wenn man die Knotenmenge V in zwei disjunkte Teilmengen U und W so aufspalten kann, dass für alle Kanten (u, w) aus der Kantenmenge E gilt: $u \in U$ und $w \in W$.

Ist der Graph G bipartit? Wenn ja, markieren Sie die Knoten von G entsprechend mit U und V . Wenn nein, geben Sie eine beliebige aber möglichst kleine Teilmenge von Knoten an, die einen Konflikt verursachen.

Begründen Sie Ihre Antwort.

c) (10 Punkte)

Kreuzen Sie an, ob die unten angegebenen Listen einer möglichen Abarbeitungsreihenfolge der Knoten bei einer Tiefen- oder Breitensuche entsprechen (*Hinweis*: Auch Mehrfachnennungen sind möglich und nur vollständig richtige Zeilen werden gewertet!).

	Tiefensuche	Breitensuche	Keines
A, J, E, H, G, D, B, F, C, I			
C, F, I, B, D, H, J, G, A, H			
A, G, J, E, H, B, D, C, I, F			
G, E, H, J, A, F, I, C, D, B			
A, B, C, D, E, F, G, H, I, J			



186.172 Algorithmen und Datenstrukturen 1 VL 4.0

2. Übungstest SS 2008

6. Juni 2008

Machen Sie die folgenden Angaben bitte in deutlicher Blockschrift:

Nachname:	<input type="text"/>	Vorname:	<input type="text"/>
Matrikelnummer:	<input type="text"/>	Studienkennzahl:	<input type="text"/>
			Anzahl abgegebener Zusatzblätter: <input type="text"/>

Legen Sie bitte Ihren Studentenausweis vor sich auf das Pult.

Sie können die Lösungen entweder direkt auf die Angabeblätter oder auf Zusatzblätter schreiben, die Sie auf Wunsch von der Aufsicht erhalten. Es ist nicht zulässig, eventuell mitgebrachtes eigenes Papier zu verwenden.

Die Verwendung von Taschenrechnern, Mobiltelefonen, Skripten, Büchern, Mitschriften, Ausarbeitungen oder vergleichbaren Hilfsmitteln ist unzulässig.

Die Arbeitszeit beträgt 55 Minuten.

	A1:	A2:	A3:	Summe:
Erreichbare Punkte:	16	16	18	50
Erreichte Punkte:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Viel Glück!

Aufgabe 1.B: Suchverfahren

(16 Punkte)

a) (4 Punkte)

- Fügen Sie folgende Werte in genau dieser Reihenfolge in einen Anfangs leeren natürlichen binären Suchbaum ein. Zeichnen Sie lediglich den resultierenden Baum, Zwischenschritte müssen nicht angegeben werden.

$\langle 59, 30, 80, 13, 40, 70, 100, 4, 35, 60, 75, 90, 101 \rangle$

- Handelt es sich bei dem resultierenden Baum um einen gültigen AVL-Baum? Begründen Sie Ihre Antwort.

b) (12 Punkte)

- Schreiben Sie detaillierten Pseudocode für eine effiziente rekursive Prozedur, die die Höhe $x.height$ und die Balancierung $x.bal$ jedes Knotens x eines natürlichen binären Suchbaumes T berechnet. Geben Sie unter Umständen notwendige Initialisierungen an sowie den ersten Aufruf mit allen Parametern, um die Rekursion zu starten. Auf die Wurzel des Baumes kann mithilfe von T zugegriffen werden. Zu jedem Knoten x werden folgende Daten gespeichert:

$x.key$: Schlüssel von x
 $x.left$: Verweis auf linkes Kind von x
 $x.right$: Verweis auf rechtes Kind von x
 $x.height$: Höhe des Teilbaumes mit Wurzel x
 $x.bal$: Balancierung des Teilbaumes mit Wurzel x

Hinweis: Die Werte $x.height$ und $x.bal$ sind zu Beginn nicht initialisiert, sondern sollen von Ihrer Prozedur berechnet werden!

- Geben Sie die Laufzeit ihres Algorithmus in Θ -Notation in Abhängigkeit der Anzahl der Knoten n an.
- Auf welchem aus der Vorlesung bekannten allgemeinen Verfahren basiert ihre Implementierung?

Aufgabe 2.B: Hashverfahren**(16 Punkte)**

a) (9 Punkte)

Gegeben ist eine Hashtabelle mit Tabellengröße $m = 7$, die Double Hashing mit der Verbesserung nach Brent benutzt mit

$$h_1(k) = k \bmod 7$$

$$h_2(k) = 2 + (k \bmod 6)$$

- Fügen Sie den Wert 42 in die folgende Tabelle ein:

0	1	2	3	4	5	6
14	8		24			

- Fügen Sie den Wert 3 in die folgende Tabelle ein:

0	1	2	3	4	5	6
14	8		24			

- Welches Problem kann beim Einfügen in die Hashtabelle bei der Verwendung der oben angegebenen Hashfunktionen auftreten? Wie müssen die Funktionen $h_1(k)$ und/oder $h_2(k)$ abgeändert werden, um dieses Problem zu beheben?

b) (7 Punkte)

Gegeben ist eine Hashtabelle mit Tabellengröße $m = 13$ und der Hashfunktion $h'(k) = k \bmod 13$. Zur Kollisionsbehandlung wird quadratisches Sondieren mit den Konstanten $c_1 = 2$ und $c_2 = 3$ verwendet.

- Fügen Sie den Wert 27 in die folgende Tabelle ein:

0	1	2	3	4	5	6	7	8	9	10	11	12
		54	16					47				37

- Fügen Sie den Wert 15 in die folgende Tabelle ein:

0	1	2	3	4	5	6	7	8	9	10	11	12
		54	16					47				37

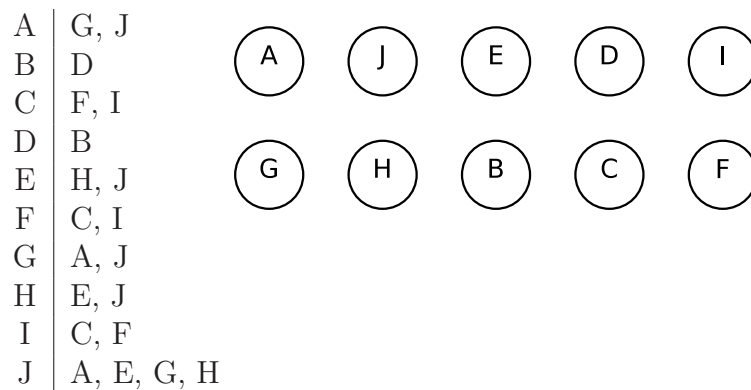
- Fügen Sie den Wert 21 in die folgende Tabelle ein:

0	1	2	3	4	5	6	7	8	9	10	11	12
		54	16					47				37

Aufgabe 3.B: Graphen

(18 Punkte)

Gegeben ist der ungerichtete Graph G in Adjazenzlistendarstellung:



a) (4 Punkte)

Zeichnen Sie den Graphen G in die gegebene Vorlage ein und geben Sie die Adjazenzmatrix von G an. Kreuzen Sie außerdem die zutreffenden Eigenschaften des Graphen G an:

enthält Kreis ist zusammenhängend ist Baum ist Wald

b) (10 Punkte)

Kreuzen Sie an, ob die unten angegebenen Listen einer möglichen Abarbeitungsreihenfolge der Knoten bei einer Tiefen- oder Breitensuche entsprechen (*Hinweis:* Auch Mehrfachnennungen sind möglich und nur vollständig richtige Zeilen werden gewertet!).

	Tiefensuche	Breitensuche	Keines
G, J, A, E, H, B, D, F, I, C			
G, H, E, J, A, I, C, F, B, D			
A, J, H, E, G, D, B, F, I, C			
A, B, C, D, E, F, G, H, I, J			
D, B, C, I, F, H, J, G, A, H			

c) (4 Punkte)

Ein ungerichteter Graph heißt *2-färbbar*, wenn man jedem Knoten genau eine der Farben F_1, F_2 zuordnen kann, sodass keinen zwei Knoten u und v , die durch eine Kante verbunden sind, die selbe Farbe zugewiesen wird.

Ist der Graph G 2-färbbar? Wenn ja, markieren Sie die Knoten von G entsprechend mit F_1 und F_2 . Wenn nein, geben Sie eine beliebige aber möglichst kleine Teilmenge von Knoten an, die einen Konflikt verursachen.

Begründen Sie Ihre Antwort.